

2017년 프로그래밍언어 개론

-03-

제출일자	2017.03.22.
이 름	정 윤 수
학 번	201302482
분 반	03

#fill out if state is 1:

```
elif state==1:
    if temp_char.isdigit():
        state = 2
    else:
        print 'ERROR'
        return None
```

문제 해결 방법

현재 state의 값이 1이라는 것은 맨 처음 문자로 '-'가 들어와서 그렇다. 처음 문자가 '-'이면 그 다음 char을 살펴보아서 '-'가 음수를 표현을 하기 위해 사용이 되었는가 특수 문자로 표현을 하기위해서 사용이 되었는지 판별을 한다. 만약 다음 char가 숫자이면 state를 2로 설정을 해주고 숫자가 아니면 ERROR를 출력을해주고 None을 반환을 해준다.

#fill out if state is 2:

```
elif state==2:
    if temp_char.isdigit(): state=2
    else:
        print 'ERROR'
        return None
```

문제 해결 방법

현재 state의 값이 2 라는 것은 첫 번째 char가 isdigit() 함수로 인하여 숫자라는 것이 판별이 되어있는 것이다. DFA에서 state가 2일 때의 전이를 살펴를 보면 현재 현재 char가 숫자라면 다시 state로 전이를 하게 되는 것을 볼 수 있다 그러므로 if문을 사용을 하여 temp_char가 숫자라면 다시 state=2를 배정을 하고 숫자가 아니라면 ERROR를 출력을 해주면서 None 값을 반환을 해준다.

#fill out if state is 3:

```
elif state==3:
    if temp_char.isdigit() or temp_char.isalpha(): state=3
    else:
        print 'ERROR'
        return None
```

문제 해결 방법

state의 값이 3이라는 것은 isalpha()함수로 인하여 첫 번째 char가 문자라는 것이 판명이 되어진 상태이다. state의 값이 3이면 DFA에서 문자 또는 숫자가 입력으로 들어오면 state가 3으로 전이가 되는 것을 볼 수있다. 그러므로 if문을 사용을 하여 temp_char.isalpha() or temp_char.isdigit()을 사용을하여 조건을 만족을 한다면 state에 3을 저장을하고 문자도 숫자도 아닌 다른 값 이라면 ERROR를 출력을 해주고 None을 반환을 해준다.

def tokenize():

```
def tokenize(self):  
    #Type is List  
    tokens=[]  
    while True :  
        tempstr = self.next_token()  
        if tempstr is not None :  
            tokens.append(tempstr)  
        else :  
            break  
    return tokens
```

문제 해결 방법

tokenize() 함수는 tokens리스트에 next_token()함수에 반환 값들을 저장을 하여 반환을 해주는 함수이다. next_token()함수로 인해 Token의 객체가 반환이 되어지는 값들을 tempstr이라는 변수를 만들어 임시로 저장을 하고. 그것을 tokens 리스트에 append()함수를 이용하여 저장을 하였다. 변수를 선언을 한 이유는 while문이 True인 상태로 계속 돌고 있어서 반복문을 탈출을 하기위해서 생성을 하였다. 먼저 tempstr를 이용하여 반환이 되어진 값을 확인을 한다. tempstr이 None값을 가지면 모든 단어들이 next_token() 함수를 수행을 한 것이다. 그러므로 break를 이용하여 while문을 탈출을 하였다.

느낀점

파이썬을 이용하여서 문자들을 parsing을 하는 과정을 알 수 있었다. 맨 처음 strip()함수를 이용하여 하여서 " "들을 제거를 해주고 split()함수를 이용하여 하여 공백을 간격으로 자르고 iter()함수를 이용하여 하여 단어들을 하나씩 판별을 하는 과정들을 거치면서 어떻게 데이터를 가공을 하는지 볼수있었다. 또한 계산이론 때 배운 DFA를 프로그래밍에 적용을 하는 것을 보고 어떤식으로 사용을 해야 하는지 알 수 있었다.